

## University of Montana ScholarWorks at University of Montana

---

Syllabi

Course Syllabi

---

9-2002

# CS 441.01: Advanced Programming - Theory and Practice

Joel Henry

*University of Montana - Missoula*, [joel.henry@umontana.edu](mailto:joel.henry@umontana.edu)

Let us know how access to this document benefits you.

Follow this and additional works at: <https://scholarworks.umt.edu/syllabi>

---

### Recommended Citation

Henry, Joel, "CS 441.01: Advanced Programming - Theory and Practice" (2002). *Syllabi*. 2697.  
<https://scholarworks.umt.edu/syllabi/2697>

This Syllabus is brought to you for free and open access by the Course Syllabi at ScholarWorks at University of Montana. It has been accepted for inclusion in Syllabi by an authorized administrator of ScholarWorks at University of Montana. For more information, please contact [scholarworks@mso.umt.edu](mailto:scholarworks@mso.umt.edu).

# Course Syllabus

## CS 441 Advanced Programming - Theory and Practice

**PREREQUISITE:** CS 335, CS 344, CS 346, CS 365 and Math 325 or consent of instructor.

### COURSE OBJECTIVES:

This is your capstone sequence in the Computer Science Department and therefore this course will extend and deepen your understanding of topics previously covered. Specifically the objectives of this course are to gain an understanding of the theory and practice involved in designing, coding, and testing software. You will examine and implement advanced data structures, algorithms, and programming techniques. You will understand design tradeoffs, software testing, maintenance and documentation issues, and program execution and memory requirements. Issues specific to team projects and project planning will be introduced. All these issues will serve as the basis for a major project in CS 442.

### INSTRUCTOR:

Name: Joel Henry  
Office: Social Sciences 411  
Office Hours: MW 9-11; Tues & Thurs 9-11 **If** the door is open I **am** available  
E-mail address: henryj@cs.umt.edu  
Phone: 243-2218

### TOPICS:

1. Challenges of design, coding, and testing.
2. Introductory programming issues.
3. Implementation issues.
4. Compilation and linking issues.
5. Software design issues.
6. Software testing, maintenance and documentation issues.
7. Real-time systems, data compression, multithreading/signaling.
8. Preparing for a major project.

### TEXTS:

**REQUIRED:** *Code Complete*, McConnell  
*Design Patterns Explained*, Shalloway

### COURSE CONTENT:

This course continues to evolve each year. These changes are to make this course the best course I can possibly provide to you. I will be using material from the Code Complete text but supplementing course materials heavily from other texts, journal articles, and personal experience. Course materials from other sources will be given to you or you can obtain them from the WWW or from reserve at the library. Please note: You may use Java or C++ in this course, and I have extensive professional computing experience which will influence this course heavily. You will not find this information in any textbook so **COME TO CLASS!**

## *Computer Science 441 - Syllabus*

### **COURSE DELIVERABLES:**

Tests (3 @ 20% each):	60%
Assignments (7):	40%

### **GRADING:**

Grading scale: 59.5 or lower F, 59.6-69.5 D, 69.6-79.5 C, 79.6-89.5 B, 89.6 – 100 A

### **LATE POLICY:**

Hand in materials:

1 day (24 hours) -	10%
2 days (48 hours) -	30%
3 days (72 hours) -	50%
4 days or more -	No thanks, I don't want it.

Presentations:

No late presentations.

### **CHEATING:**

Plagiarism will be handled harshly, as per the Student Conduct Code. You may fail the assignment or the course. MY ADVICE: Take an F rather than cheat.

### **COURSE INFORMATION:**

This course leverages the strengths of the instructor and includes course materials that meet the vision of the instructor. In short, I asked myself, "What materials would have been most useful me in a capstone course prior to graduation?" This course answers that question.

You have never had me for a course and so have no stories from other students to rely on. Here is a cliff notes version of what you might hear if you were to talk to former students of mine. I am tough but fair. I have little tolerance for trash disguised as coursework. "*The dog ate my homework*" stories do not fly. If you have a legitimate problem or obstacle to submitting assignments or taking tests, I will go the extra mile to help you. I am informal in dress and presentation. Do not let this belie the fact that I take this job and this material very seriously. You will work hard but I will be working hard along side you to provide challenging, interesting, and relevant materials. I make mistakes and have no problem admitting them. However, I am bright and experienced, and will not tolerate any disrespect. I will not disrespect you.

I firmly believe you can succeed in this course, can complete every assignment, and can master every test question. I believe you will meet my high expectations. If you are working hard at this course, I will do all I can to insure you get the grade you wish. If you attempting to do as little as possible to succeed, or just plain being lazy, you will find me indifferent to your success.

Come to class on time, just as you would if working for a prospective employer. If I am talking you are not talking. This is also a behavior I get upset with very quickly. If you want to chat, don't come to class. If you are talking when I am, you will get to leave.

## *Computer Science 441 - Syllabus*

### **Tentative Schedule:**

<b>Week</b>	<b>Tuesday</b>	<b>Thursday</b>	<b>Material</b>	<b>Assignment</b>
1	Sept. 3 – Introduction, Syllabus, Course Information, first assignment	Sept. 5 – Off – I am attending NASA project meetings – Work on Assignment 1	Chapters 1, 2, 18, and 19 McConnell	Read, Work on Assignment 1, Part A
2	Sept. 10 – Hand in Assignment 1, Part A; Collect Handouts	Sept. 12 – No class – attending NASA meetings	Chapter 3, 4, 5, and 6 McConnell	Read, Assignment 1; Part A due Sept. 10
3	Sept. 17 – Hand in Assignment 1, Part B, Designing Modules	Sept. 19 – Software Module Characteristics	Chapters 6 and 7 McConnell	Read, Assignment 1, Part B due Sept. 17
4	Sept. 24 – Return homework and lecture Design tradeoffs	Sept. 27 – Design methods and approaches	Chapters 1 and 2 Shalloway	Read, Assignment 2 due Sept. 27
5	Oct. 1 – Introduction to Testing	Oct. 3 – Introduction to Design	Chapter 3, 4, and 5 Shalloway	Read, Work on Assignment 3
6	Oct. 8 – Design Issues and Decisions	Oct. 10 – Design Tradeoffs	Review all materials	Read, Assignment 3 due Oct. 8
7	Oct. 15 – Test 1	Oct. 17 – Complex data structures – when and how	Chapters 10-12 McConnell	Read
8	Oct. 22 – Coding smartly	Oct. 24 – Conditionals and loops	Chapters 13-15 McConnell	Read, Assignment 4 due Oct. 24
9	Oct. 29 – Control structures and structuring control	Oct. 31 – The good, the bad, and the ugly	Chapters 16 and 17 McConnell	Read
10	Nov. 5 – Holiday	Nov. 7 –	Chapter 18 McConnell	Read, Assignment 5 due Nov. 7
11	Nov. 12 – Program size, breakdown, and integration	Nov. 14 – Configuration Management and Builds	Chapters 21 and 22 McConnell	Read
12	Nov. 19 – Assignment 6 presentations	Nov. 21 – Assignment 6 presentations		Assignment 6 due Nov. 19 and 21; Prepare for test 2
13	Nov. 26 – Test 2	Nov. 28 – Thanksgiving Holiday		
14	Dec. 3 – Individual preparation for team project	Dec. 5 - Team preparation for team project	Chapter 6, 7, and 8 Shalloway	
15	Dec. 10 – Project planning presentations	Dec. 12 – Project planning presentations	Chapter 9 and 10 Shalloway	Assignment 7 due Dec. 12
16	<b>Week of December 16</b>			<b>Final Exam</b>